

AperTO - Archivio Istituzionale Open Access dell'Università di Torino

## Il $\LaTeX$ come soluzione al problema dell'accesso a testi con formule da parte di disabili visivi

### This is the author's manuscript

*Original Citation:*

*Availability:*

This version is available <http://hdl.handle.net/2318/1609084> since 2017-07-05T22:24:20Z

*Terms of use:*

Open Access

Anyone can freely access the full text of works made available as "Open Access". Works made available under a Creative Commons license can be used according to the terms and conditions of said license. Use of all other works requires consent of the right holder (author or publisher) if not exempted from copyright protection by the applicable law.

(Article begins on next page)

# Il $\text{\LaTeX}$ come soluzione al problema dell'accesso a testi con formule da parte di disabili visivi

*Massimo Borsero, Nadir Murru, Alice Ruighi*

## Abstract

In questo articolo presentiamo il problema dell'accessibilità a testi con formule da parte di disabili visivi. Forniremo una breve panoramica sulle tecnologie assistive usate e su alcune attuali soluzioni al problema, evidenziandone limiti e difetti. Analizzeremo quindi l'uso del  $\text{\LaTeX}$  sia per la lettura che per la scrittura di testi con formule da parte di disabili visivi, osservando che appare come la soluzione più efficace e promettente. In particolare, vedremo quali sono le caratteristiche che lo rendono molto adatto a questo contesto, il suo insegnamento nelle scuole e il problema di generare documenti PDF accessibili a partire da questo linguaggio (problema ancora aperto).

## Abstract

In this paper we present the problem of accessibility of texts with formulae by visually impaired people. We provide a brief overview about assistive technologies and about some current solutions of this problem highlighting limits and lacks. We analyze the use of  $\text{\LaTeX}$  regarding both the reading and the writing of texts with formulae by visually impaired people, remarking that it seems to be the more efficient and promising solution. Specifically, we analyze the features that make it very suitable in this context, its teaching in the schools and the problem of creating accessible PDF documents with this language (problem still open).

## 1 Introduzione

Negli ultimi anni le tecnologie assistive hanno visto incrementare notevolmente il loro sviluppo e la loro diffusione. In particolare, nel campo della disabilità visiva, tecnologie come screen reader, barre Braille e ingranditori (nel caso di ipovisione) consentono ormai un pieno utilizzo del computer in maniera del tutto autonoma. Uno screen reader (letteralmente “lettore di schermo”) è un software dotato di sintesi vocale che legge ciò che compare nel monitor del computer e ne consente l'uso tramite comandi da tastiera. Una barra Braille è un hardware che scrive in Braille ciò che viene letto dallo screen reader.

Per mezzo di tali tecnologie, persone con disabilità visiva possono quindi fruire di materiale digitale incrementando notevolmente la possibilità, ad esempio, di intraprendere percorsi di studi, anche a livello universitario. Tuttavia, tali tecnologie

mostrano alcuni limiti quando devono maneggiare testi contenenti formule. Per una panoramica si vedano, e. g., ARCHAMBAULT *et al.* (2007) e ARMANO *et al.* (2014). Questi problemi sono dovuti principalmente al fatto che tali tecnologie sono adatte a interpretare e maneggiare strutture “unidimensionali”, ovvero testo in linea, e trovano quindi difficoltà nel caso di elementi non in linea come le formule.

La soluzione consiste nel realizzare testi con formule mediante linguaggi di marcatura (come ad esempio i linguaggi HTML e MathML). In questo modo, persone con disabilità visiva possono essere messe in condizione di leggere testi con contenuti matematici, anche se, come vedremo, a seconda del linguaggio di marcatura usato possono presentarsi comunque alcune problematiche. Un'altra questione molto importante riguarda la scrittura autonoma di testi con formule da parte di persone con disabilità visiva. Linguaggi come il MathML risultano troppo complessi per questo scopo e infatti sono stati sviluppati alcuni linguaggi di marcatura appositamente dedicati a disabili visivi, come il linguaggio LAMBDA (si veda, e. g., BERNAREGGI (2010)). Utilizzando però un linguaggio appositamente nato e sviluppato per disabili visivi si perde di vista il concetto di inclusività.

In questo contributo, presenteremo l'uso del linguaggio  $\text{\LaTeX}$  come soluzione ai problemi di accessibilità, sia per la lettura che per la scrittura, di testi contenenti formule da parte di disabili visivi. Il linguaggio  $\text{\LaTeX}$ , oltre ad essere un'ottima soluzione a queste problematiche, consente ai disabili visivi di intraprendere un percorso completamente inclusivo, in quanto il  $\text{\LaTeX}$  è il linguaggio più diffuso per la stesura di testi scientifici.

Il presente contributo è strutturato come segue. Nella sezione 2 è presentato in generale il problema dell'accessibilità a testi con formule da parte di disabili visivi, fornendo una breve panoramica su alcuni strumenti per affrontare tale problema (come il linguaggio MathML e il sistema LAMBDA), evidenziandone anche i limiti. La sezione 3 è dedicata alla discussione dell'uso del linguaggio  $\text{\LaTeX}$  come soluzione per l'accesso a testi con formule. In particolare, si discuteranno le caratteristiche che rendono il  $\text{\LaTeX}$  molto adatto in questo contesto (sezione 3.1), il suo possibile insegnamento nelle scuole (sezione 3.2), e quali sono alcuni aspetti ancora da sviluppare che ne incrementerebbero

```

<mrow>
  <mi>a</mi>
  <msup>
    <mi>x</mi>
    <mn>3</mn>
  </msup>
  <mo>+</mo>
  <mi>b</mi>
  <msup>
    <mi>x</mi>
    <mn>2</mn>
  </msup>
  <mo>+</mo>
  <mi>c</mi>
  <mi>x</mi>
  <mo>+</mo>
  <mi>d</mi>
  <mo>=</mo>
  <mn>0</mn>
</mrow>

```

FIGURE 1: Scrittura in MathML della formula  $ax^3 + bx^2 + cx + d = 0$

ulteriormente l'uso e l'utilità (sezione 3.3). Infine, nella sezione 4 si trovano le conclusioni.

## 2 Accessibilità di formule da parte di disabili visivi

Oggigiorno, persone con disabilità visiva beneficiano di screen reader (software che leggono il contenuto del monitor di un computer e ne consentono l'uso tramite comandi da tastiera) potendo così usare in maniera autonoma il computer e accedere a materiale digitale. Gli screen reader più diffusi sono NVDA (gratuito) e JAWS in ambiente Windows, VoiceOver in ambiente Mac e ORCA in ambiente Linux. Tali tecnologie assistive sono adatte a maneggiare solamente strutture in linea e quindi non hanno alcuna difficoltà a leggere il normale testo. Invece, in presenza di strutture non in linea, come immagini e formule, le prestazioni sono ancora non soddisfacenti. Per poter rendere leggibili le formule mediante screen reader è necessario che esse siano scritte mediante un linguaggio di marcatura, che provvede a una scrittura linearizzata della formula e quindi potenzialmente maneggiabile dagli screen reader.

Un linguaggio di marcatura utile in questo senso è il linguaggio MathML (Mathematical Markup Language) che permette di scrivere formule su pagine web. La scrittura di una formula mediante questo linguaggio risulta piuttosto lunga e complessa. Ad esempio una semplice equazione di terzo grado  $ax^3 + bx^2 + cx + d = 0$  è realizzata in MathML mediante la serie di comandi nella figura 1. Con questa scrittura però la formula è stata linearizzata, poiché essa corrisponde ora a una se-

rie di righe di solo testo che possono essere lette dagli screen reader. Chiaramente, la lettura delle righe nella figura 1 non rende per nulla comprensibile la formula, in quanto le istruzioni lette sono troppo lunghe e complesse. Tuttavia alcuni screen reader, quando processano una serie di righe corrispondenti a istruzioni MathML, forniscono una traduzione “al volo” di queste righe in un linguaggio più naturale. Ad esempio, nel caso della formula in esame verrà fornita la lettura “a x al cubo più b x al quadrato più c x più d uguale a zero”. Tuttavia, con l'uso del linguaggio MathML si aprono molte problematiche. Infatti, le prestazioni degli screen reader nella lettura di formule così scritte sono pesantemente influenzate da una serie di variabili: tipo di screen reader usato, tipo di browser, sistema operativo. In ambiente Mac, la lettura avviene in maniera corretta solo usando lo screen reader VoiceOver e il browser Safari. In ambiente Windows, la lettura avviene in maniera corretta solo usando lo screen reader NVDA, il browser Mozilla Firefox e con il supporto del software gratuito MathPlayer 4. In ambiente Linux, non vi è possibilità di fruire di formule scritte in MathML. Inoltre, è opportuno sottolineare che, vista la complessità di scrittura del linguaggio MathML, difficilmente un autore scrive un testo web con formule usando direttamente tale linguaggio. Solitamente viene usato un javascript chiamato MathJax che permette di scrivere le formule in maniera più comoda (anche usando il linguaggio  $\text{\LaTeX}$ ) e converte poi tali formule in MathML. Tuttavia, l'uso di questo javascript compromette l'accessibilità della formula finale. In questo modo, infatti, essa risulta accessibile solo su ambiente Windows usando lo screen reader JAWS e il browser Internet Explorer o lo screen reader NVDA e i browser Internet Explorer e Mozilla Firefox (ma in questo caso l'accesso è più difficoltoso, in quanto occorre effettuare dei passaggi preliminari non immediatamente agevoli). Un'altra problematica sull'accessibilità di formule in MathML riguarda la comodità della loro lettura. Abbiamo infatti visto che una semplice equazione di terzo grado (che ha una scrittura molto breve e compatta) viene letta dagli screen reader come “a x al cubo più b x al quadrato più c x più d uguale a zero”. La lettura di formule più complesse risulta quindi ancora più lunga e difficoltosa da interpretare. Per ovviare a questo problema esiste la possibilità, per gli utilizzatori del Braille, di usare il Braille matematico, ovvero simboli Braille che corrispondono a simboli matematici, rendendo più rapida e concisa la lettura di una formula. Tuttavia, usando il linguaggio MathML, solo per mezzo dello screen reader NVDA e del browser Mozilla Firefox è possibile ottenere la scrittura della formula in Braille matematico sull'apposita barra. Infine, osserviamo che il linguaggio MathML per la sua complessità non può essere certamente utilizzato



FIGURE 2: Scrittura in LAMBDA della formula  $\frac{x+1}{x-1}$

da disabili visivi per scrivere testi con formule. Per una panoramica sull'accessibilità di formule su web si veda BERNAREGGI e ARCHAMBAULT (2007).

Come visto, quindi, l'uso del linguaggio MathML, pur fornendo una risposta ai problemi di accessibilità delle formule, non è ottimale per molti motivi. Sono quindi state sviluppate delle soluzioni ad hoc, come il sistema LAMBDA (Linear Access to Mathematics for Braille Device and Audio-synthesis), BERNAREGGI (2010). LAMBDA è composto da un linguaggio di marcatura appositamente sviluppato, un editor dedicato e un convertitore MathML. Il linguaggio matematico in LAMBDA è stato sviluppato in modo che ogni simbolo sia tradotto “al volo” in parole. Questa traduzione è implementata nell'editor in modo che possa essere usato da disabili visivi sia per la lettura che per la scrittura della matematica. In LAMBDA, ad esempio, la formula  $\frac{x+1}{x-1}$  viene realizzata mediante la riga nella figura 2. I segnalatori grafici di inizio frazione, linea di frazione e fine frazione osservabili nella figura 2 possono essere inseriti mediante scorciatoie da tastiera e vengono tradotti automaticamente a parole o in Braille matematico da screen reader e barre Braille, permettendo così di poter fruire in maniera completa delle formule scritte con il sistema LAMBDA. Ad esempio, per iniziare una frazione è sufficiente premere Ctrl+Shift+Q e per terminarla Ctrl+K. La linea di frazione viene inserita mediante il comando Ctrl+I. Tali comandi possono essere inseriti anche usando il menù dell'editor che è completamente accessibile mediante screen reader. Inoltre, consente di svolgere esercizi di matematica in maniera accessibile. Osserviamo però che anche il sistema LAMBDA ha grossi limiti. Prima di tutto è disponibile solo su ambiente Windows. Poi, essendo un sistema nato e sviluppato per disabili visivi, non è per nulla inclusivo, in quanto gli unici conoscitori di tale linguaggio sono gli stessi utenti disabili. Quindi non è possibile per uno studente, ad esempio, trovare testi che siano stati già scritti in tale linguaggio, ma occorre effettuare delle conversioni spesso difficoltose (se non impossibili) o con risultati insoddisfacenti. Infine, non è adatto per affrontare studi di matematica a livello universitario, poiché la lista dei simboli matematici a disposizione non è completa.

Tutte le problematiche viste sull'accessibilità di formule e possibili soluzioni spingono a valutare e studiare nuove opportunità. Il linguaggio L<sup>A</sup>T<sub>E</sub>X appare essere la migliore soluzione possibile, come vedremo meglio nella prossima sezione.

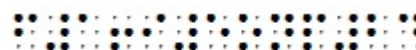


FIGURE 3: Scrittura in Braille della formula  $\frac{1+\sqrt{5}}{2}$

## 3 Il L<sup>A</sup>T<sub>E</sub>X come soluzione per l'accessibilità e l'inclusività

### 3.1 Le potenzialità del L<sup>A</sup>T<sub>E</sub>X

Il L<sup>A</sup>T<sub>E</sub>X, essendo un linguaggio di marcatura, è adatto ad essere maneggiato e letto dalle tecnologie assistive. Gli screen reader possono infatti leggere direttamente le formule scritte in L<sup>A</sup>T<sub>E</sub>X, dal momento che esse sono linearizzate come nel caso del linguaggio MathML. Nel caso del L<sup>A</sup>T<sub>E</sub>X, però, la scrittura di una formula è molto più breve, compatta e comprensibile. Anche persone che non sono disabili visivi che conoscono il L<sup>A</sup>T<sub>E</sub>X sono in grado di leggere e capire una formula direttamente dalla sua scrittura in L<sup>A</sup>T<sub>E</sub>X. Inoltre, può essere senz'altro usato da persone con disabilità visiva per scrivere testi con formule.

Esistono poi software che ne facilitano l'utilizzo e ne incrementano l'accessibilità. Il più interessante è il software BlindMath (gratuito). Esso è un editor facilitato per il L<sup>A</sup>T<sub>E</sub>X, in cui le formule possono essere inserite sia scrivendo direttamente in L<sup>A</sup>T<sub>E</sub>X sia usando un menù che consente di inserire le formule senza conoscere i comandi L<sup>A</sup>T<sub>E</sub>X. Tale menù è completamente accessibile tramite screen reader e scorciatoie da tastiera. Oltre a fornire un metodo semplificato e accessibile di inserimento delle formule, BlindMath consente anche di avere una lettura semplificata. L'utente può scegliere se leggere direttamente i comandi L<sup>A</sup>T<sub>E</sub>X corrispondenti alle formule oppure avere una lettura più “naturale” della formula. Ad esempio la formula  $\frac{1+\sqrt{5}}{2}$ , può essere letta come “\frac{1+\sqrt{5}}{2}” oppure come “inizio frazione uno più radice cinque denominatore due fine frazione”. Un'altra funzione molto utile di BlindMath è la traduzione in Braille matematico delle formule. Ad esempio, la formula precedente viene tradotta in Braille matematico come nella figura 3, che risulta un'espressione molto più compatta rispetto alla traduzione in Braille della frase “inizio frazione uno più radice cinque denominatore due fine frazione”, tradotta in Braille nella figura 4.

Utenti disabili visivi possono anche utilizzare editor classici per leggere e scrivere testi in L<sup>A</sup>T<sub>E</sub>X. Tra questi, quello caratterizzato dalla migliore accessibilità è TeXnicCenter. L'editor risulta pienamente utilizzabile mediante tecnologie assistive, ad eccezione di alcuni menù, come il menù “Build”, dove le finestre sono accessibili ma con alcune difficoltà. Inoltre, anche la finestra “Build Output”, dove in fase di compilazione vengono segnalati eventuali errori, è raggiungibile mediante screen reader con difficoltà, ma una volta raggiunta risulta

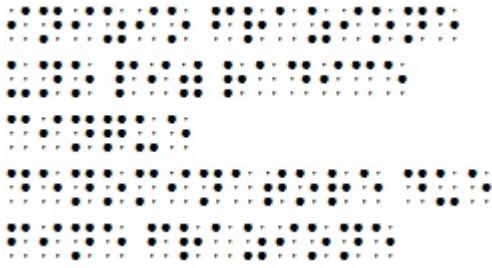


FIGURE 4: Scrittura in Braille della frase “inizio frazione uno più radice cinque denominatore due fine frazione”

accessibile. Si segnala l’esistenza del software gratuito “latex-access” che consente di attivare (a discrezione dell’utente) la lettura delle formule in linguaggio naturale direttamente nell’editor TeXnic-Center.

Gli ulteriori vantaggi dell’uso del  $\text{\LaTeX}$  rispetto a soluzioni come il MathML e LAMBDA riguardano la sua diffusione e la sua inclusività. Infatti, il  $\text{\LaTeX}$  è il linguaggio più usato per scrivere documenti scientifici, quindi è semplice reperire o convertire materiale in questo formato. Infine, il  $\text{\LaTeX}$  può essere usato e fruito in maniera accessibile su qualsiasi sistema operativo.

### 3.2 Insegnamento del $\text{\LaTeX}$ nelle scuole

Prima di affrontare brevemente la tematica dell’insegnamento del  $\text{\LaTeX}$  nelle scuole, è opportuno un rapido richiamo alle direttrici culturali e legislative entro cui questo insegnamento si inserisce:

1. In primo luogo la cosiddetta *competenza digitale*, che nella Raccomandazione del Parlamento europeo EUROPEA (2006) è definita come segue: “la competenza digitale consiste nel saper utilizzare con dimestichezza e spirito critico le tecnologie della società dell’informazione (TSI) per il lavoro, il tempo libero e la comunicazione. Essa è supportata da abilità di base nelle TIC: l’uso del computer per reperire, valutare, conservare, produrre, presentare e scambiare informazioni nonché per comunicare e partecipare a reti collaborative tramite Internet”. Tale competenza è esplicitamente richiamata nel profilo finale delle competenze al termine del primo ciclo di istruzione MIUR (2012) (ovvero a 14 anni), dove si afferma che “lo studente ha buone competenze digitali, usa con consapevolezza le tecnologie della comunicazione per ricercare ed analizzare dati e informazioni, per distinguere informazioni attendibili da quelle che necessitano di approfondimento, di controllo e di verifica e per interagire con soggetti diversi nel mondo”. Infine, la recente legge 107/15 “Buona Scuola”, di cui è possibile trovare un’analisi critica per esempio in

BRUSCHI (2015), pone tra gli obiettivi primari in carico alle istituzioni scolastiche (comma 7) lo “sviluppo delle competenze digitali degli studenti, con particolare riguardo al pensiero computazionale, all’utilizzo critico e consapevole dei social network e dei media nonché alla produzione e ai legami con il mondo del lavoro”. Inoltre (comma 56) “Al fine di sviluppare e di migliorare le competenze digitali degli studenti e di rendere la tecnologia digitale uno strumento didattico di costruzione delle competenze in generale, il Ministero dell’istruzione, dell’università e della ricerca adotta il Piano nazionale per la scuola digitale, in sinergia con la programmazione europea e regionale e con il Progetto strategico nazionale per la banda ultralarga”. Il piano ha tra i suoi fini (comma 58) la “realizzazione di attività volte allo sviluppo delle competenze digitali degli studenti, anche attraverso la collaborazione con università, associazioni, organismi del terzo settore e imprese”, ed è stato emanato alla fine del 2015 MIUR (2015) comportando un notevole incremento delle risorse dedicato all’educazione digitale.

2. In secondo luogo, il concetto di *inclusione scolastica*. Si definisce “inclusiva una scuola che permette a tutti gli alunni, tenendo conto delle loro diverse caratteristiche sociali, biologiche e culturali, non solo di sentirsi parte attiva del gruppo di appartenenza, ma anche di raggiungere il massimo livello possibile in fatto di apprendimento”, adattamento da BOOTH *et al.* (2002). Il concetto di inclusione è già presente nella celebre legge 104/92 GU1 (1992), ma è posto come fulcro del Sistema Nazionale di Istruzione e Formazione dalla legge 170/10 GU2 (2010) e successive Note applicative, dove si afferma esplicitamente che “va quindi potenziata la cultura dell’inclusione, e ciò anche mediante un approfondimento delle relative competenze degli insegnanti curricolari, finalizzata ad una più stretta interazione tra tutte le componenti della comunità educante”.

Fatte queste debite premesse, l’insegnamento del  $\text{\LaTeX}$  si inserisce perfettamente nel vasto dibattito scientifico e culturale che ha portato all’elaborazione delle leggi citate in premessa. Inoltre, a differenza di altri interventi (anche molto costosi) che in passato sono stati spinti più da mode didattiche estemporanee e che, alla prova dei fatti, hanno dimostrato scarsa efficacia (si veda, a riguardo, CALVANI (2013)), l’insegnamento del  $\text{\LaTeX}$  può essere una risposta concreta (e gratuita) a numerosi problemi di inclusione scolastica di alunni con disabilità visiva e/o con DSA (Disturbi Specifici di Apprendimento).



1. Innanzitutto, si tratta di uno strumento intrinsecamente inclusivo *tra studenti*. Infatti, a differenza del Braille o del sistema LAMBDA, conosciuti e compresi solo dai disabili visivi (e pochi educatori specializzati), può essere insegnato con efficacia *all'intera classe*. La sua compatibilità con gli screen reader, con le modalità esposte in precedenza in questo articolo, lo rende adatto all'uso degli studenti con difficoltà visive (ma anche a diverse categorie di Bisogni Educativi Speciali, come studenti dislessici o con deficit di attenzione), ma al contempo non differenzia l'uso del mezzo tecnologico tra i vari studenti della classe. Questo fatto è un grande aiuto nel consolidamento del gruppo classe, perché lo studente senza problemi di vista realizza che lo studente con disabilità visiva è in grado di apprendere e svolgere compiti con le sue stesse modalità.
2. Poi, è anche uno strumento inclusivo *tra docenti*. Non c'è nulla di peggio, per il successo educativo, di docenti dello stesso consiglio di classe che utilizzano metodologie diverse e/o confliggenti nei riguardi dello stesso studente. Invece il L<sup>A</sup>T<sub>E</sub>X, per la sua semplicità di utilizzo (almeno ad un livello elementare) può essere adoperato con profitto non solo da docenti di discipline scientifiche e tecnologiche, ma anche da tutti i docenti del consiglio di classe, e, segnatamente, dai docenti di sostegno.
3. Infine, l'insegnamento del L<sup>A</sup>T<sub>E</sub>X si colloca naturalmente nell'alveo dello sviluppo delle competenze digitali, ed è un tipo di tecnologia che, una volta appresa, resta nel bagaglio di conoscenze ed abilità dello studente, indipendentemente dalla sua particolare situazione. Il fatto poi che sia gratuito e liberamente disponibile consente a ogni studente di approfondire e diversificare quanto appreso secondo le necessità (anche lavorative) future senza vincoli economici.

Come notazione finale, si suggerisce di approfittare del particolare momento storico, in cui sono disponibili maggiori fondi per l'educazione digitale e la formazione dei docenti, per organizzare corsi ed attività di diffusione (eventualmente "formando dei formatori") della cultura e delle potenzialità del LaTeX anche, ma non solo, lungo le direttrici sopra esposte.

### 3.3 Prospettive: pacchetti per la realizzazione di PDF accessibili

Nonostante il linguaggio L<sup>A</sup>T<sub>E</sub>X sia largamente diffuso e sia ormai il linguaggio maggiormente usato per la stesura di testi scientifici, la creazione di PDF accessibili mediante il L<sup>A</sup>T<sub>E</sub>X risulta ancora un problema aperto. Sottolineiamo che si tratta di una questione molto importante, in quanto la

sua soluzione permetterebbe di diffondere documenti PDF con formule accessibili, mentre ad ora la piena accessibilità del documento può avvenire solo fruendo del sorgente L<sup>A</sup>T<sub>E</sub>X.

Un generico file PDF creato a partire da file L<sup>A</sup>T<sub>E</sub>X, infatti, viene letto dagli screen reader, in modo non del tutto soddisfacente, riscontrando alcuni problemi legati alla sua accessibilità. La nostra analisi si concentra in particolare su tre problemi: il riconoscimento dei livelli di intestazione, la lettura di una lingua diversa da quella principale usata nel testo e la lettura di formule matematiche. Si noti che mentre l'ultimo problema è proprio dei testi di natura scientifica, i primi due sono problemi di carattere più generale.

In un documento PDF le intestazioni sono essenzialmente dei TAG usati per marcare titoli di sezioni, sottosezioni, ecc. Essi sono particolarmente utili per realizzare documenti accessibili, poiché permettono di navigare in maniera efficiente e comoda un documento. Infatti, se sono presenti i livelli di intestazione, mediante uno screen reader (usando gli appositi comandi da tastiera) il lettore può passare in maniera rapida da una sezione o sottosezione all'altra del documento. Quando viene realizzato un file PDF dal L<sup>A</sup>T<sub>E</sub>X, malgrado nel file PDF venga generato un indice delle sezioni e sottosezioni del documento, non vengono create intestazioni riconoscibili dagli screen reader.

Un'altra questione importante per l'accessibilità di un documento riguarda la possibilità da parte degli screen reader di riconoscere la lingua del testo che sta processando. Infatti, se gli screen reader riconoscono la lingua in cui è scritta una frase, essa risulterà pronunciata in modo corretto, altrimenti la sua lettura risulta pesantemente compromessa. Quando viene realizzato un file PDF dal L<sup>A</sup>T<sub>E</sub>X, risulta riconoscibile da parte degli screen reader solamente la lingua principale del testo. Se nel testo sono presenti frasi in lingua differente dalla principale, esse non vengono pronunciate in maniera corretta, ma lette come se fossero scritte nella lingua principale del documento.

Infine, le formule nel file PDF realizzato a partire da un file L<sup>A</sup>T<sub>E</sub>X risultano completamente inaccessibili, ovvero non vengono lette dagli screen reader.

Alcune di queste informazioni si possono trovare per esempio in ?.

Ad oggi non paiono esistere soluzioni ottimali a queste problematiche, ma abbiamo individuato alcuni risultati soddisfacenti che si servono di alcuni pacchetti L<sup>A</sup>T<sub>E</sub>X congiuntamente ad Adobe Acrobat Pro DC (software a pagamento) e che potrebbero essere un punto di partenza per sviluppi futuri.

Un primo pacchetto interessante è `accessibility.sty` scritto da Babett Schlicht (con documentazione solo in tedesco). Un secondo pacchetto che abbiamo considerato è stato `pdfcomment.sty`, che serve principalmente per scrivere commenti che

possono essere letti sul file PDF. Entrambi i pacchetti sono stati migliorati e implementati da Andy Clifton e Ross Moore, che hanno rispettivamente creato i pacchetti `accessibility-meta.sty` e `math-sem.sty`. Tuttavia, entrambi i lavori sono ancora in via di sviluppo, non sono presenti su CTAN e i file messi a disposizione dagli autori creano dei problemi nella fase di compilazione. Per questi motivi non li abbiamo utilizzati nella nostra analisi, preferendo invece concentrarci sulle potenzialità di `accessibility.sty` e `pdfcomment.sty`.

Abbiamo analizzato e testato tali pacchetti mediante un semplice file  $\text{\LaTeX}$  di prova (con lingua principale italiana) composto da alcune sezioni e sottosezioni, una frase in lingua inglese e alcune formule. A partire da questo file di base, abbiamo creato quattro file PDF sfruttando varie combinazioni tra i due pacchetti in esame. Sono stati utilizzati Adobe Acrobat Pro DC e lo screen reader NVDA per analizzare e leggere i file PDF generati. Poiché l'accessibilità di un file è legata alle intestazioni presenti nel documento, ci siamo serviti dello strumento "tag-check" presente in Adobe Acrobat Pro DC. Esso verifica se esistono dei TAG nel documento e pertanto indica se il file PDF è dotato di una struttura.

Il primo documento PDF è stato realizzato senza l'utilizzo dei due pacchetti in esame. Da una prima analisi con tag-check esso è risultato privo di intestazioni e TAG per la lingua ("PDF con TAG: no"). Infatti, il testo del file PDF viene letto dallo screen reader NVDA (ad eccezione delle formule) senza individuare livelli di intestazione e risultando così non navigabile. Utilizzando il menù "Accessibilità" di Acrobat è possibile aggiungere dei TAG al file (in modo che tag-check fornisca un risultato positivo, ovvero "PDF con TAG: sì"), ma non ottenendo il risultato sperato dal punto di vista dell'accessibilità. Infatti, usando NVDA il documento risulta ancora non navigabile, la frase in inglese viene letta con pronuncia italiana e le formule non accessibili.

Il secondo file PDF è stato creato usando il solo pacchetto `accessibility.sty`. In questo caso, tag-check di Acrobat fornisce un risultato positivo ("PDF con TAG: sì"). Tuttavia, testando il file PDF con NVDA, si ottengono gli stessi risultati precedenti. Ora però, aggiungendo i TAG con il menù "Accessibilità" di Acrobat, si è ottenuto un documento con intestazioni riconosciute da NVDA, con cui risulta quindi possibile navigare il documento (malgrado alcune imperfezioni). La frase in inglese e la formula continuano però ad avere gli stessi problemi descritti per il file precedente.

Il terzo file PDF è stato creato usando il solo pacchetto `pdfcomment.sty`. Come nel caso del primo file, da un'analisi con tag-check risulta privo di intestazioni e l'aggiunta dei TAG con Acrobat non fornisce alcun risultato dal punto di vista

dell'accessibilità. Tuttavia, utilizzando i comandi `pdfmarkupcomment[opzioni]{testo}{commento}` e `pdftooltip{testo}{commento}`

è possibile commentare il contenuto matematico. Se si usa il comando `pdfmarkupcomment`, sul file PDF apparirà l'icona di un fumetto a fianco del testo commentato, se invece si usa il comando `pdftooltip` il commento è meno invasivo poiché non ci sono icone ad indicare dov'è il commento ed esso rimane nascosto. Inserendo tali commenti le formule risultano accessibili, in quanto gli screen reader leggono i commenti presenti nel file PDF. Questa soluzione non è però certamente ottimale. Prima di tutto NVDA, prima di leggere il contenuto del commento, quando incontra la formula la legge in maniera errata creando un po' di confusione. In secondo luogo, trattandosi di un commento (ovvero di normale testo), non è possibile in alcun modo attivare i simboli Braille per la matematica. Inoltre, nel commento sono ammesse solo normali lettere, rendendo la lettura di una formula molto lunga (non è possibile ad esempio scrivere come commento il comando  $\text{\LaTeX}$  corrispondente alla formula). Infine, tali commenti devono essere inseriti manualmente dall'autore del testo, rendendo la procedura molto lunga e tediosa.

Il quarto file PDF è stato creato usando entrambi i pacchetti, ottenendo i risultati più soddisfacenti per quanto riguarda il riconoscimento della struttura del documento e delle formule matematiche. Infatti, in questo caso (con l'aiuto di Acrobat come visto in precedenza) si ottiene un file PDF con intestazioni e con formule accessibili (anche se non in maniera del tutto soddisfacente come appena visto).

Per risolvere il problema legato agli errori di lettura dovuti alla lingua inglese, è possibile solamente agire manualmente sui TAG del file PDF mediante Acrobat. Infatti, una volta espansa tutta la struttura, si può lavorare sulle preferenze dei singoli TAG per modificare la lingua e il tipo di elemento taggato. Si noti che per modificare la lingua di una parte del testo è necessario "spezzare" il paragrafo creandone uno nuovo con il testo da modificare. Ciò è necessario perché all'interno di ogni paragrafo è possibile impostare una sola lingua di lettura.

In conclusione di questa analisi possiamo affermare che ad oggi esistono alcune parziali soluzioni ai problemi legati all'accessibilità dei documenti PDF generati con  $\text{\LaTeX}$ , ma queste sono ben lontane dall'essere ottimali. Innanzitutto, come notato precedentemente, non sono totalmente perfette nei risultati. In secondo luogo necessitano l'uso di Acrobat Reader Pro DC, software a pagamento e pertanto non fruibile da tutti. Infine, si tratta di soluzioni che richiedono un grande lavoro a posteriori. Pertanto l'obiettivo auspicato è quello di sviluppare dei pacchetti per l'accessibilità che non

richiedano l'utilizzo di strumenti esterni e che realizzino un file PDF totalmente accessibile direttamente dopo la compilazione del file L<sup>A</sup>T<sub>E</sub>X.

## 4 Conclusione

Nell'articolo abbiamo esaminato il problema dell'accessibilità di testi con formule da parte di disabili visivi. Abbiamo visto che alcune attuali soluzioni come il linguaggio MathML e il sistema LAMBDA non sono completamente soddisfacenti per vari motivi: il MathML è un linguaggio molto complesso e la sua accessibilità è soggetta a molte variabili e non sempre è ottimale; il sistema LAMBDA non è inclusivo, può essere usato solo su sistemi Windows, è poco diffuso ed è limitante per affrontare matematica ad un livello universitario. Il L<sup>A</sup>T<sub>E</sub>X invece appare essere un linguaggio completamente accessibile (sia per leggere che per scrivere testi con formule) e inclusivo (anche in vista del suo insegnamento nelle scuole), utilizzabile su tutti i sistemi operativi ed esistono software (gratuiti) che ne facilitano l'uso e ne incrementano l'accessibilità. Il suo insegnamento nelle scuole appare un passaggio ormai realizzabile nell'ottica dell'insegnamento di linguaggi di marcatura (come il linguaggio HTML) e per via, come detto, della sua inclusività. Inoltre, sottolineiamo che tra i nuovi insegnanti di matematica il linguaggio L<sup>A</sup>T<sub>E</sub>X è molto conosciuto. Si segnala a questo riguardo la tesi P.A.S. (Percorsi Abilitanti Speciali) ?, in cui è stato sperimentato con successo l'insegnamento del L<sup>A</sup>T<sub>E</sub>X a una classe di prima liceo, per facilitare gli studi di una studentessa con disabilità visiva presente nella classe. Infine, mediante il L<sup>A</sup>T<sub>E</sub>X è anche possibile generare documenti PDF parzialmente accessibili. Risulta un problema ancora aperto la generazione di PDF completamente accessibili.

## 5 Ringraziamenti

Il presente lavoro è svolto nell'ambito di una convenzione tra l'I.Ri.Fo.R./UICI (Istituto per la Ricerca, la Formazione e la Riabilitazione/Unione Italiana Ciechi e Ipovedenti) e l'Università degli Studi di Torino.

Si ringraziano la Dott.ssa Armano e la Prof.ssa Capietto per il supporto e il Prof. Andretta per averci stimolato nel presentare questo contributo.

## References

- (1992). «Legge-quadro per l'assistenza, l'integrazione sociale e i diritti delle persone handicappate (104/92)». *Gazzetta ufficiale della Repubblica italiana*, **39**.
- (2010). «Nuove norme in materia di disturbi specifici di apprendimento in ambito scolastico (170/10)». *Gazzetta ufficiale della Repubblica italiana*, **244**.
- ARCHAMBAULT, D., STOGER, B., FITZPATRICK, D. e MIESENBERGER, K. (2007). «Access to scientific content by visually impaired people». *Upgrade*, (2), p. 14.
- ARMANO, T., CAPIETTO, A., ILLENGO, M., MURRU, N. e R., R. (2014). «An overview on ict for the accessibility of scientific texts by visually impaired students». In *Conference SIREM-SIE-L*. Perugia, Italy.
- BERNAREGGI, C. (2010). «Non-sequential mathematical notations in lambda system». *Computers Helping People with Special Needs, Lecture Notes in Computer Science*, **6180**, pp. 389–395.
- BERNAREGGI, C. e ARCHAMBAULT, D. (2007). «Mathematics on the web: emerging opportunities for visually impaired people». *Proc. of the 2007 International Cross-Disciplinary Conference on web Accessibility (W4A)*, pp. 108–111.
- BOOTH, T., AINSCOW, M., BLACK-HAWKINS, K., VAUGHAN, M. e SHAW, L. (2002). «Index for inclusion». *Developing learning and participation in schools*.
- BRUSCHI, M. (2015). *La legge n.107 commentata e leggi della scuola a confronto*. Edises.
- CALVANI, A. (2013). «Le tic nella scuola: dieci raccomandazioni per i policy maker». *Formare*, **13** (4), p. 30.
- EUROPEA, U. (2006). «Raccomandazione del parlamento europeo e del consiglio del 18 dicembre 2006 relativa a competenze chiave per l'apprendimento permanente (2006/962/ce)». *Gazzetta ufficiale dell'Unione europea*, **30** (2006), pp. 10–18.
- MIUR (2012). «Indicazioni nazionali per il curriculum per la scuola dell'infanzia e del primo ciclo di istruzione».
- (2015). «Decreto n. 851 del 27 ottobre 2015».
- ▷ Massimo Borsero  
Dipartimento di Matematica “G. Peano”, Università degli Studi di Torino  
massimo.borsero@unito.it
- ▷ Nadir Murru  
Dipartimento di Matematica “G. Peano”, Università degli Studi di Torino  
nadir.murru@unito.it
- ▷ Alice Ruighi  
Dipartimento di Matematica “G. Peano”, Università degli Studi di Torino  
alice.ruighi@unito.it